

# Hot New Directions for Quasi-Monte Carlo Research in Step with Applications

Frances Y. Kuo\*

Dirk Nuyens†

## Abstract

This article provides an overview of some interfaces between the theory of quasi-Monte Carlo (QMC) methods and applications. We summarize three QMC theoretical settings: first order QMC methods in the unit cube  $[0, 1]^s$  and in  $\mathbb{R}^s$ , and higher order QMC methods in the unit cube. One important feature is that their error bounds can be independent of the dimension  $s$  under appropriate conditions on the function spaces. Another important feature is that good parameters for these QMC methods can be obtained by fast efficient algorithms even when  $s$  is large. We outline three different applications and explain how they can tap into the different QMC theory. We also discuss three cost saving strategies that can be combined with QMC in these applications. Many of these recent QMC theory and methods are developed not in isolation, but in close connection with applications.

## 1 Introduction

High dimensional computation is a new frontier in scientific computing, with applications ranging from financial mathematics such as option pricing or risk management, to groundwater flow, heat transport, and wave propagation. A tremendous amount of progress has been made in the past two decades on the theory and application of *quasi-Monte Carlo (QMC) methods* for approximating high dimensional integrals. See e.g., the classical references [64, 75] and the recent books [16, 60, 61]. One key element is the *fast component-by-component construction* [6, 66, 67, 68] which provides parameters for first order or *higher order QMC methods* [13, 16] for sufficiently smooth functions. Another key element is the careful selection of parameters called *weights* [77, 78] to ensure that the worst case errors in an appropriately weighted function space are bounded independently of the dimension. The dependence on dimension is very much the focus of the study on *tractability* [65] of multivariate problems.

We are particularly keen on the idea that new theory and methods for high dimensional computation are developed not in isolation, but in close connection with applications. The theoretical QMC convergence rates depend on the appropriate pairing between the function space and the class of QMC methods. Practitioners are free to choose the theoretical setting or pairing that is most beneficial for their applications, i.e., to achieve the best possible convergence rates under the weakest assumptions on the problems. As QMC researchers we take application problems to be our guide to develop new theory and methods as the needs arise. This article provides an overview of some interfaces between such theory and applications.

We begin in Section 2 by summarizing three theoretical settings. The first setting is what we consider to be the standard QMC setting for integrals formulated over the unit

---

\*Frances Y. Kuo (✉): School of Mathematics and Statistics, University of New South Wales, Sydney NSW 2052, Australia, [f.kuo@unsw.edu.au](mailto:f.kuo@unsw.edu.au)

†Dirk Nuyens: Department of Computer Science, KU Leuven, Celestijnenlaan 200A, 3001 Leuven, Belgium, [dirk.nuyens@cs.kuleuven.be](mailto:dirk.nuyens@cs.kuleuven.be)

cube. Here the integrand is assumed to have square-integrable mixed first derivatives, and it is paired with *randomly shifted lattice rules* [76] to achieve first order convergence. The second setting is for integration over  $\mathbb{R}^s$  against a product of univariate densities. Again the integrands have square-integrable mixed first derivatives and we use randomly shifted lattice rules to achieve first order convergence. The third setting returns to the unit cube, but considers integrands with higher order mixed derivatives and pairs them with *interlaced polynomial lattice rules* [30] which achieve higher order convergence. These three settings are discussed in more detail in [50].

Next in Section 3 we outline three applications of QMC methods: option pricing, GLMM (generalized linear mixed models) maximum likelihood, PDEs with random coefficients – all with quite different characteristics and requiring different strategies to tackle them. We explain how to match each example application with an appropriate setting from Section 2. In the option pricing application, see e.g., [2, 28, 29], none of the settings is applicable due to the presence of a *kink*. We discuss the strategy of *smoothing by preintegration* [38], which is similar to the method known as conditional sampling [1]. In the maximum likelihood application [49], the change of variables plays a crucial role in a similar way to importance sampling for Monte Carlo methods. In the PDE application, see e.g., [5, 39, 50, 73], the *uniform* and the *lognormal* cases correspond to integration over the unit cube and  $\mathbb{R}^s$ , respectively, and the two cases tap into different QMC settings. For the lognormal case we briefly contrast three ways to generate the random field: *Karhunen–Loève expansion*, *circulant embedding* [18, 33, 34, 35], and *H-matrix technique* [19, 40].

Then in Section 4 we discuss three different cost saving strategies that can be applied to all of the above applications. First, *multi-level methods* [26] restructure the required computation as a telescoping sum and tackle different levels separately to improve the overall cost versus error balance, while more general *multi-index methods* [41] allow different criteria to be considered simultaneously in a multi-index telescoping sum. Second, the *multivariate decomposition methods* [52, 58, 79] work in a similar way by making an explicit decomposition of the underlying function into functions of only subsets of the variables [59]. The third strategy is *fast QMC matrix-vector multiplication* which carries out the required computation for multiple QMC samples at the same time using an FFT [11].

We provide pointers to some software resources in Section 5 and conclude the article in Section 6 with a summary and an outlook to future work. An overview of the various components of this article is given in Figure 1.

## 2 Three Settings

Here we describe three theoretical function space settings paired with appropriate QMC methods. These three settings are also covered in [50]. Of course these three pairs are not the only possible combinations. We selected them due to our preference for constructive QMC methods that achieve the best possible convergence rates, with the implied constant independent of dimension, under the weakest possible assumptions on the integrands.

### 2.1 Setting 1: Standard QMC for the Unit Cube

For  $f$  a real-valued function defined over the  $s$ -dimensional unit cube  $[0, 1]^s$ , with  $s$  finite and fixed, we consider the integral

$$I(f) = \int_{[0,1]^s} f(\mathbf{y}) \, d\mathbf{y}. \quad (1)$$

#### Weighted Sobolev spaces

We assume in this standard setting that the integrand  $f$  belongs to a *weighted Sobolev space of smoothness one* in the unit cube  $[0, 1]^s$ . Here we focus on the *unanchored* variant in which

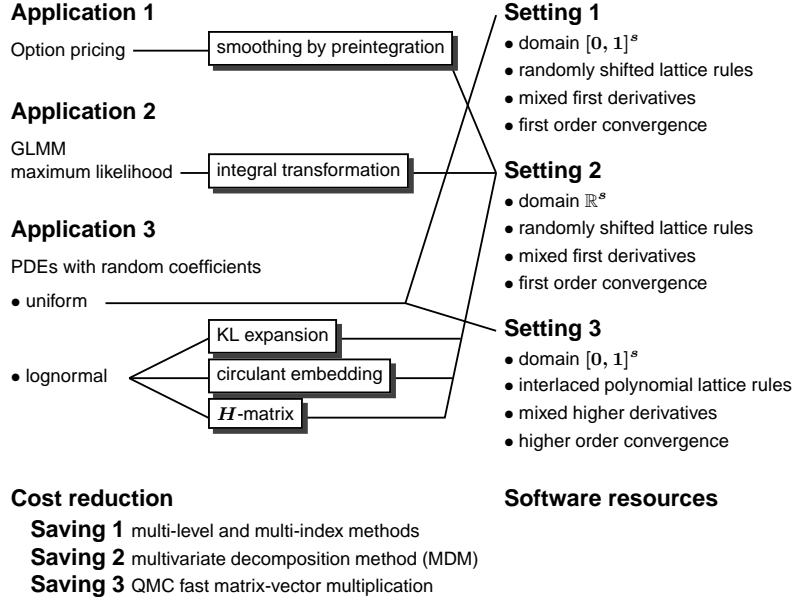


Figure 1: The connection between different components of this article.

the norm is defined by, see also [77],

$$\|f\|_{\gamma} = \left[ \sum_{\mathbf{u} \subseteq \{1:s\}} \frac{1}{\gamma_{\mathbf{u}}} \int_{[0,1]^{|\mathbf{u}|}} \left( \int_{[0,1]^{s-|\mathbf{u}|}} \frac{\partial^{|\mathbf{u}|} f(\mathbf{y})}{\partial \mathbf{y}_{\mathbf{u}}} d\mathbf{y}_{\{1:s\} \setminus \mathbf{u}} \right)^2 d\mathbf{y}_{\mathbf{u}} \right]^{1/2}, \quad (2)$$

where  $\{1:s\}$  is a shorthand notation for the set of indices  $\{1, 2, \dots, s\}$ ,  $(\partial^{|\mathbf{u}|} f)/(\partial \mathbf{y}_{\mathbf{u}})$  denotes the mixed first derivative of  $f$  with respect to the “active” variables  $\mathbf{y}_{\mathbf{u}} = (y_j)_{j \in \mathbf{u}}$ , while  $\mathbf{y}_{\{1:s\} \setminus \mathbf{u}} = (y_j)_{j \in \{1:s\} \setminus \mathbf{u}}$  denotes the “inactive” variables.

There is a weight parameter  $\gamma_{\mathbf{u}} \geq 0$  associated with each subset of variables  $\mathbf{y}_{\mathbf{u}}$  to model their relative importance. We denote the weights collectively by  $\gamma$ . Special forms of weights have been considered in the literature. *POD weights* (product and order dependent weights), arisen for the first time in [55], take the form

$$\gamma_{\mathbf{u}} = \Gamma_{|\mathbf{u}|} \prod_{j \in \mathbf{u}} \Upsilon_j,$$

which is specified by two sequences  $\Gamma_0 = \Gamma_1 = 1, \Gamma_2, \Gamma_3, \dots \geq 0$  and  $\Upsilon_1 \geq \Upsilon_2 \geq \dots > 0$ . Here the factor  $\Gamma_{|\mathbf{u}|}$  is said to be order dependent because it is determined solely by the cardinality of  $\mathbf{u}$  and not the precise indices in  $\mathbf{u}$ . The dependence of the weight  $\gamma_{\mathbf{u}}$  on the indices  $j \in \mathbf{u}$  is controlled by the product of terms  $\Upsilon_j$ . Each term  $\Upsilon_j$  in the sequence corresponds to one coordinate direction; the sequence being non-increasing indicates that successive coordinate directions become less important. Taking all  $\Gamma_{|\mathbf{u}|} = 1$  or all  $\Upsilon_j = 1$  corresponds to the weights known as *product weights* or *order dependent weights* in the literature [77, 78].

## Randomly shifted lattice rules

We pair the weighted Sobolev space with *randomly shifted lattice rules*; the complete theory can be found in [13]. Randomly shifted lattice rules approximate the integral (1) by

$$Q(f) = \frac{1}{n} \sum_{i=1}^n f(\mathbf{t}_i), \quad \mathbf{t}_i = \left\{ \frac{i \mathbf{z}}{n} + \mathbf{\Delta} \right\}, \quad (3)$$

where  $\mathbf{z} \in \mathbb{Z}^s$  is known as the *generating vector*,  $\mathbf{\Delta}$  is a *random shift* drawn from the uniform distribution over  $[0, 1]^s$ , and the braces indicate that we take the fractional parts of each component in a vector.

A randomly shifted lattice rule provides an unbiased estimate of the integral, i.e.,  $\mathbb{E}[Q(f)] = I(f)$ , where the expectation is taken with respect to the random shift  $\mathbf{\Delta}$ . Its quality is determined by the choice of the generating vector  $\mathbf{z}$ . By analyzing the quantity known as *shift-averaged worst case error*, it is known that good generating vectors can be obtained using a *CBC construction* (*component-by-component construction*), determining the components of  $\mathbf{z}$  one at a time sequentially, to achieve nearly  $\mathcal{O}(n^{-1})$  convergence rate which is optimal in the weighted Sobolev space of smoothness one, and the implied constant in the big  $\mathcal{O}$  bound can be independent of  $s$  under appropriate conditions on the weights  $\gamma$ .

More precisely, if  $n$  is a power of 2 then we know that the CBC construction yields the root-mean-square error bound, for all  $\lambda \in (1/2, 1]$ ,

$$\sqrt{\mathbb{E}[|I(f) - Q(f)|^2]} \leq \left( \frac{2}{n} \sum_{\emptyset \neq \mathbf{u} \subseteq \{1:s\}} \gamma_{\mathbf{u}}^{\lambda} [\vartheta(\lambda)]^{|\mathbf{u}|} \right)^{1/(2\lambda)} \|f\|_{\gamma}, \quad (4)$$

where  $\vartheta(\lambda) := 2\zeta(2\lambda)/(2\pi^2)^{\lambda}$ , with  $\zeta(a) := \sum_{k=1}^{\infty} k^{-a}$  denoting the Riemann zeta function. A similar result holds for general  $n$ . The best rate of convergence clearly comes from choosing  $\lambda$  close to  $1/2$ , but the advantage is offset by the fact that  $\zeta(2\lambda) \rightarrow \infty$  as  $\lambda \rightarrow (1/2)_+$ .

## Choosing the weights

To apply this abstract theory to a given practical integrand  $f$ , we need to first obtain an estimate of the norm  $\|f\|_{\gamma}$ . Remember that at this stage we do not yet know how to choose the weights  $\gamma_{\mathbf{u}}$ . Assuming that bounds on the mixed first derivatives in (2) can be obtained so that

$$\|f\|_{\gamma} \leq \left( \sum_{\mathbf{u} \subseteq \{1:s\}} \frac{B_{\mathbf{u}}}{\gamma_{\mathbf{u}}} \right)^{1/2}, \quad (5)$$

we can substitute (5) into (4) and then, with  $\lambda$  fixed but unspecified at this point and  $A_{\mathbf{u}} = [\vartheta(\lambda)]^{|\mathbf{u}|}$ , we choose the weights  $\gamma_{\mathbf{u}}$  to minimizing the product

$$C_{\gamma} := \left( \sum_{\mathbf{u} \subseteq \{1:s\}} \gamma_{\mathbf{u}}^{\lambda} A_{\mathbf{u}} \right)^{1/(2\lambda)} \left( \sum_{\mathbf{u} \subseteq \{1:s\}} \frac{B_{\mathbf{u}}}{\gamma_{\mathbf{u}}} \right)^{1/2}.$$

Elementary calculus leads us to conclude that we should take

$$\gamma_{\mathbf{u}} := \left( \frac{B_{\mathbf{u}}}{A_{\mathbf{u}}} \right)^{1/(1+\lambda)}, \quad (6)$$

which yields

$$C_{\gamma} = \left( \sum_{\mathbf{u} \subseteq \{1:s\}} A_{\mathbf{u}}^{1/(1+\lambda)} B_{\mathbf{u}}^{\lambda/(1+\lambda)} \right)^{(1+\lambda)/(2\lambda)}.$$

We then specify a value of  $\lambda$ , as close to  $1/2$  as possible, to ensure that  $C_\gamma$  can be bounded independently of  $s$ . This in turn determines the theoretical convergence rate which is  $\mathcal{O}(n^{-1/(2\lambda)})$ .

The chosen weights  $\gamma_u$  are then fed into the CBC construction to produce generating vectors for randomly shifted lattice rules that achieve the desired theoretical error bound for this integrand. This strategy for determining weights was first considered in [55].

*Fast* CBC constructions (using FFT) can produce generating vectors for an  $n$ -point rule in  $s$  dimensions in  $\mathcal{O}(sn \log n)$  operations in the case of product weights [66], and in  $\mathcal{O}(sn \log n + s^2 n)$  operations in the case of POD weights [54]. Note that these are considered to be pre-computation costs. The actual cost for generating the points on the fly is  $\mathcal{O}(sn)$  operations, no worse than Monte Carlo simulations. Strategies to improve on the computational cost of approximating the integral are discussed in Section 4.

The CBC construction yields a lattice rule which is extensible in dimension  $s$ . We can also construct *lattice sequences* which are extensible or embedded in the number of points  $n$ , at the expense of increasing the implied constant in the error bound [6, 17, 45, 46].

## 2.2 Setting 2: QMC Integration over $\mathbb{R}^s$

QMC approximation to an integral which is formulated over the Euclidean space  $\mathbb{R}^s$  can be obtained by first mapping the integral to the unit cube as follows:

$$\begin{aligned} I(f) &= \int_{\mathbb{R}^s} f(\mathbf{y}) \prod_{j=1}^s \phi(y_j) d\mathbf{y} = \int_{[0,1]^s} f(\Phi^{-1}(\mathbf{w})) d\mathbf{w} \\ &\approx \frac{1}{n} \sum_{i=1}^n f(\Phi^{-1}(\mathbf{t}_i)) = Q(f). \end{aligned} \quad (7)$$

(With a slight abuse of notation we have reused  $I(f)$  and  $Q(f)$  from the previous subsection for integration over  $\mathbb{R}^s$  in this subsection.) Here  $\phi$  can be *any general univariate probability density function*, and  $\Phi^{-1}$  denotes the component-wise application of the inverse of the cumulative distribution function corresponding to  $\phi$ . Note that in many practical applications we need to first apply some clever transformation to convert the integral into the above form; some examples are discussed in Section 3. The transformed integrand  $f \circ \Phi^{-1}$  arising from practical applications typically does not belong to the Sobolev space defined over the unit cube due to the integrand being unbounded near the boundary of the cube, or because the mixed derivatives of the transformed integrand do not exist or are unbounded. Thus the theory in the preceding subsection generally does not apply in practice. Some theory for QMC on singular integrands is given in [70].

We summarize here a special weighted space setting in  $\mathbb{R}^s$  for which randomly shifted lattice rules have been shown to achieve nearly the optimal convergence rate of order one [57, 63]. The norm in this setting is given by

$$\begin{aligned} \|f\|_\gamma &= \left[ \sum_{\mathbf{u} \subseteq \{1:s\}} \frac{1}{\gamma_{\mathbf{u}}} \int_{\mathbb{R}^{|\mathbf{u}|}} \left( \int_{\mathbb{R}^{s-|\mathbf{u}|}} \frac{\partial^{|\mathbf{u}|} f}{\partial \mathbf{y}_{\mathbf{u}}}(\mathbf{y}) \left( \prod_{j \in \{1:s\} \setminus \mathbf{u}} \phi(y_j) \right) d\mathbf{y}_{\{1:s\} \setminus \mathbf{u}} \right)^2 \right. \\ &\quad \left. \times \left( \prod_{j \in \mathbf{u}} \varpi_j^2(y_j) \right) d\mathbf{y}_{\mathbf{u}} \right]^{1/2}. \end{aligned} \quad (8)$$

Comparing (8) with (2), apart from the difference that the integrals are now over the unbounded domain, there is a probability density function  $\phi$  as well as additional *weight functions*  $\varpi_j$  which can be chosen to moderate the tail behavior of the mixed derivatives of  $f$ .

The convergence results for the CBC construction of randomly shifted lattice rules in this general setting depend on the choices of  $\phi$  and  $\varpi_j$ . For  $n$  a power of 2, the root-mean-square

error bound takes the form, for all  $\lambda \in (1/(2r), 1]$ ,

$$\sqrt{\mathbb{E} [|I(f) - Q(f)|^2]} \leq \left( \frac{2}{n} \sum_{\emptyset \neq u \subseteq \{1:s\}} \gamma_u^\lambda \prod_{j \in u} \vartheta_j(\lambda) \right)^{1/(2\lambda)} \|f\|_\gamma,$$

with  $r$  (appearing in the applicable lower bound on  $\lambda$ ) and  $\vartheta_j(\lambda)$  depending on  $\phi$  and  $\varpi_j$ , see [63, Theorem 8]. Some special cases have been analyzed:

- See [32, Theorem 15] or [50, Theorem 5.2] for  $\phi(y) = \phi_{\text{nor}}(y) = \exp(-y^2/2)/\sqrt{2\pi}$  being the standard normal density and  $\varpi_j^2(y_j) = \exp(-2\alpha_j |y_j|)$  with  $\alpha_j > 0$ .
- See [50, Theorem 5.3] for  $\phi = \phi_{\text{nor}}$  and  $\varpi_j^2(y_j) = \exp(-\alpha y_j^2)$  with  $\alpha < 1/2$ .
- See [74, Theorem 2] for  $\phi$  being a logistic, normal, or Student density and  $\varpi_j = 1$ .

To apply this abstract theory to a practical integral over  $\mathbb{R}^s$ , it is important to realize that the choice of  $\phi$  can be tuned as part of the process of transformation to express the integral in the form (7). (This point will become clearer when we describe the maximum likelihood application in Subsection 3.2.) Then the choice of weight functions  $\varpi_j$  arises as part of the process to obtain bounds on the norm of  $f$ , as in (5). (This point will become clearer when we describe the PDE application in Subsection 3.3.) Finally we can choose the weights  $\gamma_u$  as in (6) but now with  $A_u = \prod_{j \in u} \vartheta_j(\lambda)$  for the appropriate  $\vartheta_j(\lambda)$  corresponding to the choice of  $\phi$  and  $\varpi_j$ . The choice of density  $\phi$ , weight functions  $\varpi_j$ , and weight parameters  $\gamma_u$  then enter the CBC construction to obtain the generating vector of good randomly shifted lattice rules that can achieve the theoretical error bound for this integrand.

In practice, it may well be that the weights  $\gamma_u$  obtained in this way are not sensible because we were working with theoretical upper bounds on the error that may be too pessimistic. It may already be so in the standard setting of the previous subsection, but is more pronounced in the setting for  $\mathbb{R}^s$  due to the additional complication associated with the presence of  $\phi$  and  $\varpi_j$ .

### 2.3 Setting 3: Smooth Integrands in the Unit Cube

Now we return to the integration problem over the unit cube (1) and outline a weighted function space setting from [10] for smooth integrands of order  $\alpha$ . The norm is given by

$$\|f\|_\gamma = \sup_{u \subseteq \{1:s\}} \sup_{\mathbf{y}_v \in [0,1]^{|v|}} \frac{1}{\gamma_u} \sum_{v \subseteq u} \sum_{\tau_{u \setminus v} \in \{1:\alpha\}^{|u \setminus v|}} \left| \int_{[0,1]^{s-|v|}} (\partial^{(\alpha_v, \tau_{u \setminus v}, \mathbf{0})} f)(\mathbf{y}) d\mathbf{y}_{\{1:s\} \setminus v} \right|. \quad (9)$$

Here  $(\alpha_v, \tau_{u \setminus v}, \mathbf{0})$  denotes a multi-index  $\nu$  with  $\nu_j = \alpha$  for  $j \in v$ ,  $\nu_j = \tau_j$  for  $j \in u \setminus v$ , and  $\nu_j = 0$  for  $j \notin u$ . We denote the  $\nu$ -th partial derivative of  $f$  by  $\partial^\nu f = (\partial^{|\nu|} f) / (\partial_{y_1}^{\nu_1} \partial_{y_2}^{\nu_2} \dots \partial_{y_s}^{\nu_s})$ .

This function space setting can be paired with *interlaced polynomial lattice rules* [30, 31] to achieve higher order convergence rates in the unit cube. A *polynomial lattice rule* [64] is similar to a lattice rule (see (3) without the random shift  $\Delta$ ), but instead of a generating vector of integers we have a generating vector of polynomials, and thus the regular multiplication and division are replaced by their polynomial equivalents. We omit the technical details here. An interlaced polynomial lattice rule with  $n = 2^m$  points in  $s$  dimensions with interlacing factor  $\alpha$  is obtained by taking a polynomial lattice rule in  $\alpha s$  dimensions and then interlacing the bits from every successive  $\alpha$  dimensions to yield one dimension. More explicitly, for  $\alpha = 3$ , given three coordinates  $x = (0.x_1 x_2 \dots x_m)_2$ ,  $y = (0.y_1 y_2 \dots y_m)_2$  and  $z = (0.z_1 z_2 \dots z_m)_2$  we interlace their bits to obtain  $w = (0.x_1 y_1 z_1 x_2 y_2 z_2 \dots x_m y_m z_m)_2$ .

An interlaced polynomial lattice rule with interlacing factor  $\alpha \geq 2$ , with irreducible modulus polynomial of degree  $m$ , and with  $n = 2^m$  points in  $s$  dimensions, can be constructed

by a CBC algorithm such that, for all  $\lambda \in (1/\alpha, 1]$ ,

$$|I(f) - Q(f)| \leq \left( \frac{2}{n} \sum_{\emptyset \neq u \subseteq \{1:s\}} \gamma_u^\lambda [\vartheta_\alpha(\lambda)]^{|u|} \right)^{1/\lambda} \|f\|_\gamma,$$

where  $\vartheta_\alpha(\lambda) := 2^{\alpha\lambda(\alpha-1)/2}([1 + 1/(2^{\alpha\lambda} - 2)]^\alpha - 1)$ . This result can be found in [50, Theorem 5.4], which was obtained from minor adjustments of [10, Theorem 3.10].

Given a practical integrand  $f$ , if we can estimate the corresponding integrals involving the mixed derivatives in (9), then we can choose the weights  $\gamma_u$  so that every term in the supremum is bounded by a constant, say,  $c$ . This strategy in [10] led to a new form of weights called *SPOD weights* (*smoothness-driven product and order dependent weights*); they take the form

$$\gamma_u = \sum_{\nu_u \in \{1:\alpha\}^{|u|}} \Gamma_{|\nu_u|} \prod_{j \in u} \Upsilon_j(\nu_j).$$

If the weights  $\gamma$  are SPOD weights, then the fast CBC construction of the generating vector has cost  $\mathcal{O}(\alpha s n \log n + \alpha^2 s^2 n)$  operations. If the weights  $\gamma$  are product weights, then the CBC algorithm has cost  $\mathcal{O}(\alpha s n \log n)$  operations.

### 3 Three Applications

Integrals over  $\mathbb{R}^s$  often arise from practical applications in the form of multivariate expected values

$$\mathbb{E}_\rho[q] = \int_{\mathbb{R}^s} q(\mathbf{y}) \rho(\mathbf{y}) d\mathbf{y}, \quad (10)$$

where  $q$  is some quantity of interest which depends on a vector  $\mathbf{y} = (y_1, \dots, y_s)$  of parameters or variables in  $s$  dimensions, and  $\rho$  is some multivariate probability density function describing the distribution of  $\mathbf{y}$ , *not necessarily a product of univariate functions* as we assumed in (7), and so we need to make an appropriate transformation to apply our theory. Below we discuss three motivating applications with quite different characteristics, and we will explain how to make use of the different settings in Section 2.

#### 3.1 Application 1: Option Pricing

Following the Black–Scholes model, integrals arising from option pricing problems take the general form (10), with

$$q(\mathbf{y}) = \max(\mu(\mathbf{y}), 0) \quad \text{and} \quad \rho(\mathbf{y}) = \frac{\exp(-\frac{1}{2}\mathbf{y}^T \Sigma^{-1} \mathbf{y})}{\sqrt{(2\pi)^s \det(\Sigma)}},$$

where the variables  $\mathbf{y} = (y_1, \dots, y_s)^T$  correspond to a discretization of the underlying Brownian motion over a time interval  $[0, T]$ , and the covariance matrix has entries  $\Sigma_{ij} = (T/s) \min(i, j)$ . For example, in the case of an *arithmetic average Asian call option* [2, 28, 29], the payoff function  $q$  depends on the smooth function  $\mu(\mathbf{y}) = (1/s) \sum_{j=1}^s S_{t_j}(\mathbf{y}) - K$  which is the difference between the average of the asset prices  $S_{t_j}$  at the discrete times and the strike price  $K$ .

The widely accepted strategy to rewrite these option pricing integrals from the form (10) to the form (7) with product densities is to take a factorization  $\Sigma = AA^T$  and apply a change of variables  $\mathbf{y} = A\mathbf{y}'$ . This yields an integral of the form (7) with

$$f(\mathbf{y}') = q(A\mathbf{y}') \quad \text{and} \quad \phi = \phi_{\text{nor}}.$$

The choice of factorization therefore determines the function  $f$ . For example,  $A$  can be obtained through Cholesky factorization (commonly known as the *standard construction*; in

this case it is equivalent to generating the Brownian motions sequentially in time), through *Brownian bridge construction* [4], or eigenvalue decomposition sometimes called the *principal component construction* [2]. Note that in practice these factorizations are not carried out explicitly due to the special form of the covariance matrix. In fact, they can be computed in  $\mathcal{O}(s)$ ,  $\mathcal{O}(s)$  and  $\mathcal{O}(s \log s)$  operations, respectively [28].

The success of QMC for option pricing cannot be explained by most existing theory due to the *kink* in the integrand induced by the maximum function. However, for some factorizations it is shown in [37] that all *ANOVA terms* of  $f$  are smooth, with the exception of the highest order term. This hints at a *smoothing by preintegration* strategy, where a coordinate with some required property is chosen, say  $y_k$ , and we integrate out this one variable (either exactly or numerically with high precision) to obtain a function in  $s - 1$  variables

$$P_k(f) := \int_{-\infty}^{\infty} f(\mathbf{y}) \phi_{\text{nor}}(y_k) dy_k.$$

Under the right conditions (e.g., integrating with respect to  $y_1$  in the case of the principal components construction), this new function is smooth and belongs to the function space setting of Subsection 2.2 (with one less variable) [38]. This strategy is related to the method known as *conditional sampling* [1].

### 3.2 Application 2: Maximum Likelihood

Another source of integrands which motivated recent developments in the function space setting of Subsection 2.2 is a class of generalized linear mixed models (GLMM) in statistics, as examined in [49, 57, 74]. A specific example of the Poisson likelihood time series model considered in these papers involves an integral of the form (10), with

$$q(\mathbf{y}) = \prod_{j=1}^s \frac{\exp(\tau_j(\beta + y_j) - e^{\beta + y_j})}{\tau_j!} \quad \text{and} \quad \rho(\mathbf{y}) = \frac{\exp(-\frac{1}{2}\mathbf{y}^T \Sigma^{-1} \mathbf{y})}{\sqrt{(2\pi)^s \det(\Sigma)}}.$$

Here  $\beta \in \mathbb{R}$  is a model parameter,  $\tau_1, \dots, \tau_s \in \{0, 1, \dots\}$  are the count data, and  $\Sigma$  is a Toeplitz covariance matrix with  $\Sigma_{ij} = \sigma^2 \kappa^{|i-j|} / (1 - \kappa^2)$ , where  $\sigma^2$  is the variance and  $\kappa \in (-1, 1)$  is the autoregression coefficient.

An obvious way to rewrite this integral in the form (7) with product densities is to factorize  $\Sigma$  as discussed in the previous subsection for the option pricing applications, but this would yield a very spiky function  $f$ . Instead, the strategy developed in [49] recenters and rescales the exponent  $T(\mathbf{y})$  of the product  $q(\mathbf{y})\rho(\mathbf{y}) =: \exp(T(\mathbf{y}))$  as follows:

1. Find the unique stationary point  $\mathbf{y}^*$  satisfying  $\nabla T(\mathbf{y}^*) = 0$ .
2. Determine the matrix  $\Sigma^* = (-\nabla^2 T(\mathbf{y}^*))^{-1}$  which describes the convexity of  $T$  around the stationary point.
3. Factorise  $\Sigma^* = A^* A^{*\top}$ .
4. Apply a change of variables  $\mathbf{y} = A^* \mathbf{y}' + \mathbf{y}^*$ .
5. Multiply and divide the resulting integrand by the product  $\prod_{j=1}^s \phi(y'_j)$  where  $\phi$  is any univariate density (not necessarily the normal density).

These steps then yield an integral of the form (7) with

$$f(\mathbf{y}') = \frac{c \exp(T(A^* \mathbf{y}' + \mathbf{y}^*))}{\prod_{j=1}^s \phi(y'_j)}$$

for some scaling constant  $c > 0$ . Note that the choice of  $A^*$  and  $\phi$  determines  $f$ .

The paper [74] provides careful estimates of the norm of the resulting integrand  $f$  in the setting of Subsection 2.2 corresponding to three different choices of density  $\phi$ , with the



weight functions taken as  $\varpi_j = 1$ , and gives the formula for the weight parameters  $\gamma_u$  that minimize the overall error bound.

These GLMM problems are extremely challenging not only for QMC but also in general the tools are still lacking. There is still lots of room to develop new QMC methods and theory for these problems.

### 3.3 Application 3: PDEs with Random Coefficients

Our third application is motivated by fluid flow through a porous medium, typically modelled using Darcy's Law, with random coefficients. A popular toy problem is the elliptic PDE with a random coefficient [5, 39, 73]

$$-\nabla \cdot (a(\mathbf{x}, \omega) \nabla u(\mathbf{x}, \omega)) = \kappa(\mathbf{x}) \quad \text{for } \mathbf{x} \in D \subset \mathbb{R}^d \text{ and almost all } \omega \in \Omega,$$

with  $d \in \{1, 2, 3\}$ , subject to homogeneous Dirichlet boundary conditions. The coefficient  $a(\mathbf{x}, \omega)$  is assumed to be a random field over the spatial domain  $D$  (e.g., representing the permeability of a porous material over  $D$ ), and  $\Omega$  is the probability space. The goal is to compute the expected values  $\mathbb{E}[G(u)]$  of some bounded linear functional  $G$  of the solution  $u$  over  $\Omega$ .

For practical reasons it is often assumed that  $a(\mathbf{x}, \omega)$  is a *lognormal random field*, that is,  $a(\mathbf{x}, \omega) = \exp(Z(\mathbf{x}, \omega))$ , where  $Z(\mathbf{x}, \omega)$  is a Gaussian random field with a prescribed mean and covariance function. This is known as the *lognormal case*. However, researchers often analyze a simpler model known as the *uniform case*.

#### The uniform case

In the uniform case, we consider the parametric PDE

$$-\nabla \cdot (a(\mathbf{x}, \mathbf{y}) \nabla u(\mathbf{x}, \mathbf{y})) = \kappa(\mathbf{x}) \quad \text{for } \mathbf{x} \in D \subset \mathbb{R}^d, \quad (11)$$

together with

$$a(\mathbf{x}, \mathbf{y}) = a_0(\mathbf{x}) + \sum_{j \geq 1} y_j \psi_j(\mathbf{x}), \quad (12)$$

where the parameters  $y_j$  are independently and uniformly distributed on the interval  $[-\frac{1}{2}, \frac{1}{2}]$ , and we assume that  $0 < a_{\min} \leq a(\mathbf{x}, \mathbf{y}) \leq a_{\max} < \infty$  for all  $\mathbf{x}$  and  $\mathbf{y}$ .

A (single-level) strategy for approximating  $\mathbb{E}[G(u)]$  is as follows:

1. Truncate the infinite sum in (12) to  $s$  terms.
2. Solve the PDE using finite element methods with meshwidth  $h$ .
3. Approximate the resulting  $s$ -dimensional integral using QMC with  $n$  points.

So the error is a sum of truncation error, discretization error, and quadrature error.

For the QMC quadrature error in Step 3, we have the integral (1) with

$$f(\mathbf{y}) = G(u_h^s(\cdot, \mathbf{y} - \frac{1}{2})),$$

where  $u_h^s$  denotes the finite element solution of the truncated problem, and the subtraction by  $\frac{1}{2}$  takes care of the translation from the usual unit cube  $[0, 1]^s$  to  $[-\frac{1}{2}, \frac{1}{2}]^s$ . By differentiating the PDE (11), we can obtain bounds on the mixed derivatives of the PDE solution with respect to  $\mathbf{y}$ . This leads to bounds on the norm (2) of the integrand  $f$  and so we can apply the theoretical setting of Subsection 2.1 to obtain up to first order convergence for QMC. Under appropriate assumptions and with first order finite elements, we can prove that the total error for the above 3-step strategy is of order [55]

$$\mathcal{O}(s^{-2(1/p_0-1)} + h^2 + n^{-\min(1/p_0-1/2, 1-\delta)}), \quad \delta \in (0, \frac{1}{2}),$$

where  $p_0 \in (0, 1)$  should be as small as possible while satisfying  $\sum_{j \geq 1} \|\psi_j\|_{L^\infty}^{p_0} < \infty$ . This part is presented as a step-by-step tutorial in the article [51] from this volume.

The bounds on the derivatives of the PDE with respect to  $\mathbf{y}$  also allow us to obtain bounds on the norm (9) and so we can also apply the theoretical setting of Subsection 2.3 to obtain higher order convergence [10]. Specifically, the  $\mathcal{O}(n^{-\min(1/p_0-1/2, 1-\delta)})$  term can be improved to  $\mathcal{O}(n^{-1/p_0})$ . Also the  $\mathcal{O}(h^2)$  term can be improved by using higher order finite elements. See [50, 51] for more details.

### The lognormal case with Karhunen–Loève expansion

In the lognormal case, we have the same parametric PDE (11), but now we use the *Karhunen–Loève expansion* (*KL expansion*) of the Gaussian random field (in the exponent) to write

$$a(\mathbf{x}, \mathbf{y}) = a_0(\mathbf{x}) \exp \left( \sum_{j \geq 1} y_j \sqrt{\mu_j} \xi_j(\mathbf{x}) \right),$$

where  $a_0(\mathbf{x}) > 0$ , the  $\mu_j$  are real, positive and non-increasing in  $j$ , the  $\xi_j$  are orthonormal in  $L_2(D)$ , and the parameters  $y_j \in \mathbb{R}$  are standard  $\mathcal{N}(0, 1)$  random variables. Truncating the infinite series in  $a(\mathbf{x}, \mathbf{y})$  to  $s$  terms and solving the PDE with a finite element method as in the uniform case, we have now an integral of the form (7) with

$$f(\mathbf{y}) = G(u_h^s(\cdot, \mathbf{y})) \quad \text{and} \quad \phi = \phi_{\text{nor}}.$$

One crucial step in the analysis of [32] is to choose suitable weight functions  $\varpi_j$  so that the function  $f$  has a finite and indeed small norm (8), so that the theoretical setting of Subsection 2.2 can be applied. Again see [50, 51] for more details.

In this lognormal case with KL expansion (and also the uniform case), the cost per sample of the random field is  $\mathcal{O}(sM)$  operations, where  $M$  is the number of finite element nodes. This dominates the cost in evaluating the integrand function under the assumption that assembling the stiffness matrix to solve the PDE (which depends on the random field) is higher than the cost of the PDE solve which is  $\mathcal{O}(M \log M)$ . When  $s$  is large the cost of sampling the random field can be prohibitive, and this is why the following alternative strategies emerged.

### The lognormal case with circulant embedding

Since we have a Gaussian random field we can actually sample the random field exactly on any set of  $M$  spatial points. This leads to an integral of the form (10) with (assuming the field has zero mean)

$$q(\mathbf{y}) = G(u_h^s(\cdot, \mathbf{y})) \quad \text{and} \quad \rho(\mathbf{y}) = \frac{\exp(-\frac{1}{2} \mathbf{y}^T R^{-1} \mathbf{y})}{\sqrt{(2\pi)^s \det(R)}},$$

where  $R$  is an  $M \times M$  covariance matrix, and initially we have  $s = M$ . (Note the subtle abuse of notation that the second argument in  $u_h^s(\mathbf{x}, \cdot)$  has a different meaning to the KL case, in the sense that there the covariance is already built in.) This integral can be transformed into the form (7) with a factorization  $R = AA^T$  and a change of variables  $\mathbf{y} = A\mathbf{y}'$ , as in the option pricing example, to obtain

$$f(\mathbf{y}') = G(u_h^s(\cdot, A\mathbf{y}')) \quad \text{and} \quad \phi = \phi_{\text{nor}}.$$

The advantage of this discrete formulation is that there is no error arising from the truncation of the KL expansion. However, the direct factorization and matrix-vector multiplication require  $\mathcal{O}(M^3)$  operations which can be too costly when  $M$  is large.

The idea of *circulant embedding* [18, 33, 34, 35] is to sample the random field on a regular grid and to embed the covariance matrix of these points into a larger  $s \times s$  matrix which is *nested block circulant with circulant blocks*, so that FFTs can be used to reduce the per sample cost to  $\mathcal{O}(s \log s)$  operations. Values of the random field at the finite element quadrature nodes can be obtained by interpolation. Note that this turns the problem into an even higher dimensional integral, and we can have  $s \gg M$ . For this strategy to work we need to use regular spatial grid points to sample the field and a stationary covariance function (i.e., the covariance depends only on the relative distance between points). An additional difficulty is to ensure positive definiteness of the extended matrix; this is studied in [34].

### The lognormal case with $H$ -matrix technique

Another approach for the discrete matrix formulation of the lognormal case is to first approximate  $R$  by an  $H$ -matrix [40] and make use of  $H$ -matrix techniques to compute the matrix-vector multiplication with the square-root of this  $H$ -matrix at essentially linear cost  $\mathcal{O}(M)$ . Two iterative methods have been proposed in [19] to achieve this (one is based on a variant of the *Lanczos iteration* and the other on the *Schultz iteration*), with full theoretical justification for the error incurred in the  $H$ -matrix approximation. An advantage of this approach over circulant embedding is that it does not require the spatial grid to be regular nor that the covariance be stationary.

### Other developments

A different QMC analysis for the lognormal case has been considered in [42]. QMC for holomorphic equations was considered in [14], and for Bayesian inversion in [8, 72]. Recently there is also QMC analysis developed for the situation where the functions in the expansion of  $a(\mathbf{x}, \mathbf{y})$  have local support, see [22, 43, 48].

## 4 Three Cost Saving Strategies

In this section we discuss the basic ideas of three different kinds of cost saving strategies that can be applied to QMC methods, without going into details. Actually, the circulant embedding and  $H$ -matrix technique discussed in the previous section can also be considered as cost saving strategies. These strategies are not mutually exclusive, and it may be possible to mix and match them to benefit from compound savings.

### 4.1 Saving 1: Multi-level and Multi-index

The *multi-level* idea [26] is easy to explain in the context of numerical integration. Suppose that there is a sequence  $(f_\ell)_{\ell \geq 0}$  of approximations to an integrand  $f$ , with increasing accuracy and cost as  $\ell$  increases, such that we have the telescoping sum

$$f = \sum_{\ell=0}^{\infty} (f_\ell - f_{\ell-1}), \quad f_{-1} := 0.$$

For example, the different  $f_\ell$  could correspond to different number of time steps in option pricing, different number of mesh points in a finite element solve for PDE, different number of terms in a KL expansion, or a combination of aspects. A multi-level method for approximating the integral of  $f$  is

$$A_{\text{ML}}(f) = \sum_{\ell=0}^L Q_\ell(f_\ell - f_{\ell-1}),$$

where the parameter  $L$  determines the number of *levels*, and for each level we apply a different quadrature rule  $Q_\ell$  to the difference  $f_\ell - f_{\ell-1}$ .

The integration error (in this simple description with deterministic quadrature rules) satisfies

$$|I(f) - A_{\text{ML}}(f)| \leq \underbrace{|I(f) - I_L(f_L)|}_{\leq \varepsilon/2} + \underbrace{\sum_{\ell=0}^L |(I_\ell - Q_\ell)(f_\ell - f_{\ell-1})|}_{\leq \varepsilon/2}.$$

For a given error threshold  $\varepsilon > 0$ , the idea (as indicated by the underbraces) is that we choose  $L$  to ensure that the first term (the truncation error) on the right-hand side is  $\leq \varepsilon/2$ , and we specify parameters for the quadrature rules  $Q_\ell$  so that the second term (the quadrature error) is also  $\leq \varepsilon/2$ . The latter can be achieved with a Lagrange multiplier argument to minimize cost subject to the given error threshold. Our hope is that the successive differences  $f_\ell - f_{\ell-1}$  will become smaller with increasing  $\ell$  and therefore we would require less quadrature points for the more costly higher levels.

The *multi-index* idea [41] generalizes this from a scalar level index  $\ell$  to a vector index  $\boldsymbol{\ell}$  so that we can vary a number of different aspects (e.g., spatial/temporal discretization) simultaneously and independently of each other. It makes use of the sparse grid concept so that the overall cost does not blow up with respect to the dimensionality of  $\boldsymbol{\ell}$ , i.e., the number of different aspects being considered. A simple example is that we use different finite element meshwidths for different spatial coordinates. This is equivalent to applying sparse finite element methods within a multilevel algorithm, see the article [27] in this volume.

Multi-level and multi-index extensions of QMC methods for the applications from Section 3 include e.g., [12, 29, 53, 56, 71].

## 4.2 Saving 2: Multivariate Decomposition Method

In the context of numerical integration, the *multivariate decomposition method* (MDM) [25, 52, 58, 79] makes use of a decomposition of the integrand  $f$  of the form

$$f = \sum_{|\mathbf{u}| < \infty} f_{\mathbf{u}},$$

where the sum is over all finite subsets  $\mathbf{u} \subset \{1, 2, \dots\}$  and each function  $f_{\mathbf{u}}$  depends only on the integration variables with indices in the set  $\mathbf{u}$ . Then MDM takes the form

$$A_{\text{MDM}}(f) = \sum_{\mathbf{u} \in \mathcal{A}} Q_{\mathbf{u}}(f_{\mathbf{u}})$$

where  $\mathcal{A}$  is known as the *active set* of subsets of indices, and for each  $\mathbf{u}$  in the active set we apply a different quadrature rule  $Q_{\mathbf{u}}$  to  $f_{\mathbf{u}}$ .

Analogously to the multi-level idea, the error of MDM satisfies

$$|I(f) - A_{\text{MDM}}(f)| \leq \underbrace{\sum_{\mathbf{u} \notin \mathcal{A}} |I_{\mathbf{u}}(f_{\mathbf{u}})|}_{\leq \varepsilon/2} + \underbrace{\sum_{\mathbf{u} \in \mathcal{A}} |(I_{\mathbf{u}} - Q_{\mathbf{u}})(f_{\mathbf{u}})|}_{\leq \varepsilon/2},$$

where we choose the active set  $\mathcal{A}$  to ensure that the truncation error is  $\leq \varepsilon/2$ , and we use a Lagrange multiplier argument to specify parameters for the quadrature rules so that the quadrature error is also  $\leq \varepsilon/2$ . Our hope is that, although the cardinality of the active set  $\mathcal{A}$  might be huge (e.g., tens of thousands), the cardinality of the individual subsets  $\mathbf{u} \in \mathcal{A}$  might be relatively small (e.g., at most 8 or 10), and therefore we transfer the problem into that of solving a large number of low dimensional integrals.

There are many important considerations in the implementation of MDM [24]. First, we need to decide on how to decompose the integrand  $f$  so that values of the functions  $f_{\mathbf{u}}$  can

be computed. One obvious choice is known as the *anchored decomposition* which can be computed via the explicit formula [59]

$$f_u(\mathbf{y}_u) = \sum_{\mathbf{v} \subseteq u} (-1)^{|u|-|\mathbf{v}|} f(\mathbf{y}_{\mathbf{v}}; \mathbf{a}), \quad (13)$$

where  $\mathbf{a}$  is an *anchor* and  $(\mathbf{y}_{\mathbf{v}}; \mathbf{a})$  denotes a vector obtained from  $\mathbf{y}$  by replacing the component  $y_j$  with the corresponding component  $a_j$  when the index  $j$  does not belong to the subset  $\mathbf{v}$ . (This is similar to the well-known ANOVA decomposition which, however, involves integrals that cannot be computed in practice.) Second, we need to specify and construct the active set  $\mathcal{A}$  and have an efficient data structure to store the sets for later traversing. Third, we need to explore nestedness or embedding in the quadrature rules, taking into account the sum in (13) and develop efficient ways to reuse function evaluations.

### 4.3 Saving 3: Fast QMC Matrix-vector Multiplication

There is a certain structure in some QMC methods that can allow for fast matrix-vector multiplication using FFT. This structure has been exploited in the fast CBC construction of lattice rules and polynomial lattice rules [66]. We now explain how this same structure can also be used in more general circumstances [11].

For notational convenience, we denote all QMC points  $\mathbf{t}_i$  as row vectors in this subsection. Given an arbitrary matrix  $A$ , suppose we want to

$$\text{compute } \mathbf{y}_i A \text{ for all } i = 1, \dots, n,$$

with the row vectors  $\mathbf{y}_i = \chi(\mathbf{t}_i)$ , where  $\chi$  denotes an arbitrary univariate function that is applied to every component of the QMC point  $\mathbf{t}_i$ . Typically we have  $\mathbf{t}_n \equiv \mathbf{t}_0 = \mathbf{0}$  so we can leave it out. Consider for simplicity the case  $n$  is prime and suppose we can write

$$Y := \begin{bmatrix} \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_{n-1} \end{bmatrix} = C P$$

where  $C$  is an  $(n-1) \times (n-1)$  circulant matrix, while  $P$  is a matrix containing a single 1 in each column and 0 everywhere else. Then we can compute  $Y\mathbf{a}$  in  $\mathcal{O}(n \log n)$  operations for any column  $\mathbf{a}$  of  $A$ .

The desired factorization  $Y = CP$  is possible if we have deterministic lattice points or deterministic polynomial lattice points, and if we apply the inverse cumulative distribution function mapping or tent transform [7, 15, 44]. However, it does *not* work with random shifting, scrambling [69], or interlacing. This strategy can be used to generate normally distributed points with a general covariance matrix (no need for stationarity as in circulant embedding), solving PDEs with uniform random coefficients, or solving PDEs with lognormal random coefficients involving finite element quadratures.

## 5 Software Resources

We provide some software resources for the practical application of QMC methods:

- *The Magic Point Shop*: a collection of QMC point generators and generating vectors.  
<https://people.cs.kuleuven.be/~dirk.nuyens/qmc-generators/>
- *Fast component-by-component constructions*: a collection of software routines for fast CBC constructions of generating vectors.  
<https://people.cs.kuleuven.be/~dirk.nuyens/fast-cbc/>

- *QMC4PDE*: accompanying software package for the survey [50] on using QMC methods for parametrized PDE problems.  
<https://people.cs.kuleuven.be/~dirk.nuyens/qmc4pde/>
- *A practical guide to QMC methods*: a non-technical introduction of QMC methods with software demos.  
<https://people.cs.kuleuven.be/~dirk.nuyens/taiwan/>

## 6 Summary and Outlook

In this article we summarized three QMC theoretical settings: randomly shifted lattice rules achieving first order convergence in the unit cube and in  $\mathbb{R}^s$ , and interlaced polynomial lattice rules achieving higher order convergence in the unit cube. One important feature is that the error bound can be independent of the dimension under appropriate conditions on the weights. Another important feature is that these QMC methods can be constructed by fast CBC algorithms.

We outlined three different applications and explained how they can be pre-processed to make use of the different theory. We also discussed three cost saving strategies that can be combined with QMC in these applications.

This paper is not meant to be a comprehensive survey on QMC methods. There are of course many other significant developments on QMC methods and their applications. For example, we did not discuss *tent transformation* (also known as the baker’s transform), which can yield second order convergence for randomly shifted rules or first order convergence for deterministic lattice rules [7, 15, 44]. We also did not discuss *scrambling* [69], which is a well-known randomization method that can potentially improve the convergence rates by an extra half order.

For the future we would like to see QMC in new territories, to tackle a significantly wider range of more realistic problems. Some emerging new application areas of QMC include e.g., Bayesian inversion [8, 72], stochastic wave propagation [20, 21], quantum field theory [3, 47], and neutron transport [23, 36].

Looking ahead into future QMC developments, what would be on the top of our wish list? We would very much like to have a “*Setting 4*” where we have QMC methods that achieve higher order convergence in  $\mathbb{R}^s$ , with error bounds that are independent of  $s$ , and for which fast constructions are possible. This open problem has seen some partial solutions [9, 62] but there is more to be done!

**Acknowledgement.** The authors acknowledge the financial supports from the Australian Research Council (FT130100655 and DP150101770) and the KU Leuven research fund (OT:3E130287 and C3:3E150478).

## References

- [1] Achtsis, N., Cools, R., Nuyens, D.: Conditional sampling for barrier option pricing under the Heston model, in: Monte Carlo and Quasi-Monte Carlo Methods 2012 (J. Dick, F. Y. Kuo, G. Peters, and I. H. Sloan, eds), Springer, Berlin, pp. 253–269 (2013)
- [2] Acworth, P., Broadie, M., Glasserman, P.: A comparison of some Monte Carlo and quasi-Monte Carlo techniques for option pricing, in: Monte Carlo and quasi-Monte Carlo methods 1996 (P. Hellekalek, G. Larcher, H. Niederreiter, and P. Zinterhof, eds.), Springer, Berlin, pp. 1–18 (1998)

- [3] Ammon, A., Hartung, T., Jansen, K., Leövey, H., Vollmer, J.: On the efficient numerical solution of lattice systems with low-order couplings, *Comput. Phys. Commun.* **198**, 71–81 (2016)
- [4] Caflisch, R.E., Morokoff, W., Owen, A.B.: Valuation of mortgage backed securities using Brownian bridges to reduce effective dimension, *J. Comput. Finance* **1**, 27–46 (1997)
- [5] Cohen, A., DeVore, R.: Approximation of high-dimensional parametric PDEs, *Acta Numer.* **24**, 1–159 (2015)
- [6] Cools, R., Kuo, F.Y., Nuyens, D.: Constructing embedded lattice rules for multivariate integration, *SIAM J. Sci. Comput.* **28**, 2162–2188 (2006)
- [7] Cools, R., Kuo, F.Y., Nuyens, D., Suryanarayana, G.: Tent-transformed lattice rules for integration and approximation of multivariate non-periodic functions, *J. Complexity* **36**, 166–181 (2016)
- [8] Dick, J., Gantner, R.N., Le Gia, Q.T., Schwab, Ch.: Higher order Quasi-Monte Carlo integration for Bayesian estimation (in review)
- [9] Dick, J., Irrgeher, Ch., Leobacher, G., Pillichshammer, F.: On the optimal order of integration in Hermite spaces with finite smoothness (in review)
- [10] Dick, J., Kuo, F.Y., Le Gia, Q.T., Nuyens, D., Schwab, Ch.: Higher order QMC Galerkin discretization for parametric operator equations, *SIAM J. Numer. Anal.* **52**, 2676–2702 (2014)
- [11] Dick, J., Kuo, F.Y., Le Gia, Q.T., Schwab, Ch.: Fast QMC matrix-vector multiplication, *SIAM J. Sci. Comput.* **37**, A1436–A1450 (2015)
- [12] Dick, J., Kuo, F.Y., Le Gia, Q.T., Schwab, Ch.: Multi-level higher order QMC Galerkin discretization for affine parametric operator equations, *SIAM J. Numer. Anal.*, **54**, 2541–2568 (2016)
- [13] Dick, J., Kuo, F.Y., Sloan, I.H.: High-dimensional integration: the Quasi-Monte Carlo way, *Acta Numer.* **22**, 133–288 (2013)
- [14] Dick, J., Le Gia, Q.T., Schwab, Ch.: Higher order Quasi-Monte Carlo integration for holomorphic, parametric operator equations, *SIAM/ASA Journal on Uncertainty Quantification*, **4**, 48–79 (2016)
- [15] Dick, J., Nuyens, D., Pillichshammer, F.: Lattice rules for nonperiodic smooth integrands, *Numer. Math.* **126**, 259–291 (2014)
- [16] Dick, J., Pillichshammer, F.: *Digital Nets and Sequences*, Cambridge University Press (2010)
- [17] Dick, J., Pillichshammer, F., Waterhouse, B.J.: The construction of good extensible rank-1 lattices, *Math. Comp.* **77**, 2345–2374 (2008)
- [18] Dietrich, C.R., Newsam, G.H.: Fast and exact simulation of stationary Gaussian processes through circulant embedding of the covariance matrix, *SIAM J. Sci. Comput.* **18**, 1088–1107 (1997)
- [19] Feischl, M., Kuo, F.Y., Sloan, I.H.: Fast random field generation with  $H$ -matrices (in review)
- [20] Ganesh, M., Hawkins, S.C.: A high performance computing and sensitivity analysis algorithm for stochastic many-particle wave scattering, *SIAM J. Sci. Comput.* **37**, A1475–A1503 (2015)
- [21] Ganesh, M., Kuo, F.Y., Sloan, I.H.: Quasi-Monte Carlo finite element methods for stochastic heterogeneous wave propagation models (in preparation)
- [22] Gantner, R.N., Herrmann, L., Schwab, Ch.: Quasi-Monte Carlo integration for affine-parametric, elliptic PDEs: local supports imply product weights (in review)

- [23] Gilbert, A.D., Graham, I.G., Kuo, F.Y., Scheichl, R., Sloan, I.H.: Quasi-Monte Carlo theory for an eigenproblem with a random coefficient (in preparation)
- [24] Gilbert, A.D., Kuo, F.Y., Nuyens, D., Wasilkowski, G.W.: Efficient implementation of the multivariate decomposition method (in preparation)
- [25] Gilbert, A.D., Wasilkowski, G.W.: Small superposition dimension and active set construction for multivariate integration under modest error demand, *J. Complexity* **42**, 94–109 (2017)
- [26] Giles, M.B.: Multilevel Monte Carlo methods, *Acta Numer.* **24**, 259–328 (2015)
- [27] Giles, M.B., Kuo, F.Y., Sloan, I.H.: Combining sparse grids, multilevel MC and QMC for elliptic PDEs with random coefficients (submitted to this volume)
- [28] Giles, M.B., Kuo, F.Y., Sloan, I.H., Waterhouse, B.J.: Quasi-Monte Carlo for finance applications, *ANZIAM J.* **50**, C308–C323 (CTAC2008) (2008)
- [29] Giles, M.B., Waterhouse, B.J.: Multilevel quasi-Monte Carlo path simulation. *Radon Series Comp. Appl. Math.* **8**, 1–18 (2009)
- [30] Goda, T.: Good interlaced polynomial lattice rules for numerical integration in weighted Walsh spaces, *J. Comput. Appl. Math.* **285**, 279–294 (2015)
- [31] Goda, T., Dick, J.: Construction of interlaced scrambled polynomial lattice rules of arbitrary high order, *Found. Comput. Math.* **15**, 1245–1278 (2015)
- [32] Graham, I.G., Kuo, F.Y., Nichols, J.A., Scheichl, R., Schwab, Ch., Sloan, I.H.: QMC FE methods for PDEs with log-normal random coefficients, *Numer. Math.* **131**, 329–368 (2015)
- [33] Graham, I.G., Kuo, F.Y., Nuyens, D., Scheichl, R., Sloan, I.H.: Quasi-Monte Carlo methods for elliptic PDEs with random coefficients and applications, *J. Comput. Phys.* **230**, 3668–3694 (2011)
- [34] Graham, I.G., Kuo, F.Y., Nuyens, D., Scheichl, R., Sloan, I.H.: Analysis of circulant embedding methods for sampling stationary random fields (in preparation)
- [35] Graham, I.G., Kuo, F.Y., Nuyens, D., Scheichl, R., Sloan, I.H.: Circulant embedding with QMC – analysis for elliptic PDE with lognormal coefficients (in preparation)
- [36] Graham, I.G., Parkinson, M.J., Scheichl, R.: Modern Monte Carlo variants for uncertainty quantification in neutron transport (in review)
- [37] Griebel, M., Kuo, F.Y., Sloan, I.H.: The smoothing effect of integration in  $\mathbb{R}^d$  and the ANOVA decomposition, *Math. Comp.* **82** (2013), 383–400; and the note in *Math. Comp.* **86**, 1847–1854 (2017)
- [38] Griewank, A., Kuo, F.Y., Leövey, H., Sloan, I.H.: High dimensional integration of kinks and jumps – smoothing by preintegration (in review)
- [39] Gunzburger, M., Webster, C., Zhang, G.: Stochastic finite element methods for partial differential equations with random input data, *Acta Numer.* **23**, 521–650 (2014)
- [40] Hackbusch, W.: *Hierarchical matrices: algorithms and analysis*, Springer, Heidelberg (2015)
- [41] Haji-Ali, A.L., Nobile, F., Tempone, R.: Multi-index Monte Carlo: when sparsity meets sampling, *Numer. Math.* **132**, 767–806 (2016)
- [42] Harbrecht, H., Peters, M., Siebenmorgen, M.: On the quasi-Monte Carlo method with Halton points for elliptic PDEs with log-normal diffusion, *Math. Comp.* **86**, 771–797 (2017)
- [43] Herrmann, L., Schwab, Ch.: QMC integration for lognormal-parametric, elliptic PDEs: local supports imply product weights (in review)



- [44] Hickernell, F.J.: Obtaining  $O(N^{-2+\epsilon})$  convergence for lattice quadrature rules, in: Monte Carlo and Quasi-Monte Carlo Methods 2000 (K. T. Fang, F. J. Hickernell, and H. Niederreiter, eds.), Springer, Berlin, pp. 274–289 (2002)
- [45] Hickernell, F.J., Hong, H.S., LÉcuyer, P., Lemieux, C.: Extensible lattice sequences for quasi-Monte Carlo quadrature, *SIAM J. Sci. Comput.* **22**, 1117–1138 (2000)
- [46] Hickernell, F.J., Niederreiter, H.: The existence of good extensible rank-1 lattices, *J. Complexity* **19**, 286–300 (2003)
- [47] Jansen, K., Leövey, H., Griewank, A., Müller-Preussker, M.: Quasi-Monte Carlo methods for lattice systems: a first look, *Comput. Phys. Commun.* **185**, 948–959 (2014)
- [48] Kazashi, Y.: Quasi-Monte Carlo integration with product weights for elliptic PDEs with log-normal coefficients (in review)
- [49] Kuo, F.Y., Dunsmuir, W.D.M., Sloan, I.H., Wand, M.P., Womersley, R.S.: Quasi-Monte Carlo for highly structured generalised response models *Methodol. Comput. App.* **10**, 239–275 (2008)
- [50] Kuo, F.Y., Nuyens, D.: Application of quasi-Monte Carlo methods to elliptic PDEs with random diffusion coefficients – a survey of analysis and implementation, *Found. Comput. Math.* **16**, 1631–1696 (2016)
- [51] Kuo, F.Y., Nuyens, D.: Application of quasi-Monte Carlo methods to PDEs with random coefficients – an overview and tutorial (to appear in this volume)
- [52] Kuo, F.Y., Nuyens, D., Plaskota, L., Sloan, I.H., Wasilkowski, G.W.: Infinite-dimensional integration and the multivariate decomposition method, *J. Comput. Appl. Math.* **326**, 217–234 (2017)
- [53] Kuo, F.Y., Scheichl, R., Schwab, Ch., Sloan, I.H., Ullmann, E.: Multilevel Quasi-Monte Carlo methods for lognormal diffusion problems, *Math. Comp.* **86**, 2827–2860 (2017)
- [54] Kuo, F.Y., Schwab, Ch., Sloan, I.H.: Quasi-Monte Carlo methods for high dimensional integration: the standard weighted-space setting and beyond, *ANZIAM J.* **53**, 1–37 (2011)
- [55] Kuo, F.Y., Schwab, Ch., Sloan, I.H.: Quasi-Monte Carlo finite element methods for a class of elliptic partial differential equations with random coefficient, *SIAM J. Numer. Anal.* **50**, 3351–3374 (2012)
- [56] Kuo, F.Y., Schwab, Ch., Sloan, I.H.: Multi-level quasi-Monte Carlo finite element methods for a class of elliptic partial differential equations with random coefficient, *Found. Comput. Math.* **15**, 411–449 (2015)
- [57] Kuo, F.Y., Sloan, I.H., Wasilkowski, G.W., Waterhouse, B.J.: Randomly shifted lattice rules with the optimal rate of convergence for unbounded integrands, *J. Complexity* **26**, 135–160 (2010)
- [58] Kuo, F.Y., Sloan, I.H., Wasilkowski, G.W., Woźniakowski, H.: Liberating the dimension, *J. Complexity* **26**, 422–454 (2010)
- [59] Kuo, F.Y., Sloan, I.H., Wasilkowski, G.W., Woźniakowski, H.: On decompositions of multivariate functions, *Math. Comp.* **79**, 953–966 (2010)
- [60] Lemieux, C.: *Monte Carlo and Quasi-Monte Carlo Sampling*, Springer, New York (2009)
- [61] Leobacher, G., Pillichshammer, F.: *Introduction to Quasi-Monte Carlo Integration and Applications*, Springer (2014)
- [62] Nguyen, D.T.P., Nuyens, D.: Multivariate integration over  $\mathbb{R}^s$  with exponential rate of convergence, *J. Comput. Appl. Math.* **315**, 327–342 (2017)

- [63] Nichols, J.A., Kuo F.Y.: Fast CBC construction of randomly shifted lattice rules achieving  $\mathcal{O}(N^{-1+\delta})$  convergence for unbounded integrands in  $\mathbb{R}^s$  in weighted spaces with POD weights, *J. Complexity* **30**, 444–468 (2014)
- [64] Niederreiter, H.: *Random Number Generation and Quasi-Monte Carlo Methods*, SIAM, Philadelphia (1992)
- [65] Novak, E., Woźniakowski, H.: *Tractability of Multivariate Problems, II: Standard Information for Functionals*, European Mathematical Society, Zürich (2010)
- [66] Nuyens, D.: The construction of good lattice rules and polynomial lattice rules, in: *Uniform Distribution and Quasi-Monte Carlo Methods* (P. Kritzer, H. Niederreiter, F. Pillichshammer, A. Winterhof, eds.), *Radon Series on Computational and Applied Mathematics Vol. 15*, De Gruyter, pp. 223–256 (2014)
- [67] Nuyens, D., Cools, R.: Fast algorithms for component-by-component construction of rank-1 lattice rules in shift-invariant reproducing kernel Hilbert spaces, *Math. Comp.* **75**, 903–920 (2006)
- [68] Nuyens, D., Cools, R.: Fast component-by-component construction of rank-1 lattice rules with a non-prime number of points, *J. Complexity* **22**, 4–28 (2006)
- [69] Owen, A.B.: Scrambled net variance for integrals of smooth functions, *Ann. Statist.* **25**, 1541–1562 (1997)
- [70] Owen, A.B.: Halton sequences avoid the origin, *SIAM Rev.* **48**, 487–503 (2006)
- [71] Robbe, P., Nuyens, D., Vandewalle, S.: A multi-index quasi-Monte Carlo algorithm for lognormal diffusion problems, *SIAM J. Sci. Comput.* (to appear)
- [72] Scheichl, R., Stuart, A., Teckentrup, A.L.: Quasi-Monte Carlo and multilevel Monte Carlo methods for computing posterior expectations in elliptic inverse problems, *SIAM/ASA Journal on Uncertainty Quantification* **5**, 493–518 (2017)
- [73] Schwab, Ch., Gittelson, C.J.: Sparse tensor discretizations of high-dimensional parametric and stochastic PDEs, *Acta Numer.* **20**, 291–467 (2011)
- [74] Sinescu, V., Kuo, F.Y., Sloan, I.H.: On the choice of weights in a function space for quasi-Monte Carlo methods for a class of generalised response models in statistics, in: *Monte Carlo and Quasi-Monte Carlo Methods 2012* (J. Dick, F. Y. Kuo, G. Peters, and I. H. Sloan, eds), Springer Verlag, Heidelberg, pp. 631–647 (2013)
- [75] Sloan, I.H., Joe, S.: *Lattice Methods for Multiple Integration*, Oxford University Press, Oxford (1994)
- [76] Sloan, I.H., Kuo, F.Y., Joe, S.: Constructing randomly shifted lattice rules in weighted Sobolev spaces, *SIAM J. Numer. Anal.* **40**, 1650–1665 (2002)
- [77] Sloan, I.H., Wang, X., Woźniakowski, H.: Finite-order weights imply tractability of multivariate integration, *J. Complexity* **20**, 46–74 (2004)
- [78] Sloan, I.H., Woźniakowski, H.: When are quasi-Monte Carlo algorithms efficient for high-dimensional integrals?, *J. Complexity* **14**, 1–33 (1998)
- [79] Wasilkowski, G.W.: On tractability of linear tensor product problems for  $\infty$ -variate classes of functions, *J. Complexity* **29**, 351–369 (2013)